**Промова**

Добридень, шановна комісія та всі присутні в цій аудиторії. Я хочу представити Вашій увазі дипломну роботу бакалавра на тему «Реалізація мовного процесора у середовищі Python». Ми не намагаємось створити конкуренцію компаніям-гігантам, виробникам компіляторів, де над кожною стадією працює декілька десятків програмістів. Метою роботи є показати основні підходи до реалізації семантичного аналізу і генерації проміжного коду і відкрити нюанси, що трапляються під час реалізації вищевказаних стадій і показати методи їх вирішення. Бо, як відомо, 2 вищевказані стадії майже не описані в літературі, і підходи до їх реалізації є комерційною таємницею цих фірм. Важливо параметризувати мовний процесор, щоб можна було модифікувати граматику мови без втручання у вихідний код програмного продукту. Постановка задачі складається з декількох пунктів: **читаю слайд**

Перше питання, яке постало перед початком створення диплому, є вибір мови реалізації. **Читаю 2 слайди**

Як відомо, загальна схема роботи компілятора включає наступні стадії **читаю слайд.**

Розглянемо 1 стадію, лексичний аналіз. Задачею ЛА є перевірити вірність усіх слів у вхідному потоці. Апаратом виділення лексем із вхідного потоку були вибрані регулярні вирази, ось як виглядає представлення класів лексем на мові регулярних виразів. **слайд** Ці метадані винесені в окремий файл конфігурації, в будь-який момент можна відредагувати правила для ідентифікаторів, констант, літералів. Алгоритм виділення лексем виглядає ось як **читаю слайд** Кожна виділена лексема в процесі роботи лексичного аналізатора додається до таблиці атрибутів, **читаю слайд** Подальші стадії роботи мов проц. постійно звертаються за інформацією про кожну лексему до таблиці атрибутів.

Наступною стадією мовного процесора синтаксичний аналіз. Задачею його є перевірити правильність порядку лексем у вхідному файлі. **читаю слайд** Для зберігання інформації у файлі був вибраний формат YAML через цілий ряд переваг **читаю слайд** Отже, граматика у вигляді РБНФ зберігається у форматі YAML. Ось як переводиться правила граматики з РБНФ у YAML **читаю слайд**

3 стадією реалізації мов проц. є сем А. Задачею семантичного аналізу розпізнавання смислу виразів і контроль правильності цього смислу. Для реалізації сем А був вибраний апарат семантичних дій. По тексту граматики проставляються позначення дій, які треба виконувати під час аналізу конкретного слова. В основному це перевірка сумісності типів, і побудова абстрактного дерева розбору. **читаю 2 слайди** Для побудови АДР і полегшення подальшої генерації коду розроблена система атрибутних класів **читаю слайд** АДР має приблизно вигляд як **на слайді**. Кожен атрибутний клас має покажчик на наступну структуру.

Ми перейшли до стадії генерації проміжного коду. Ця стадія полегшує майбутню генерацію машинного коду, оскільки видає лінійну структуру виконання команд. В якості проміжної мови використовуємо мову тетрад. **читаю слайд**

В якості висновків можна відзначити **читаю слайд**